NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
GODDARD SPACE FLIGHT CENTER

INSTITUTE FOR SPACE STUDIES
NEW YORK, N. Y.

MAIL ADDRESS
INSTITUTE FOR SPACE STUDIES
2880 BROADWAY
NEW YORK, N. Y. 10025

TELEPHONE
UNIVERSITY 6-3600

FEB 1 1974

29 January 1974

Professor Joshua Lederberg
Department of Genetics
School of Medicine
Stanford University Medical Center
Stanford, California 94305

Dear Professor Lederberg:

Your letter of January 23 raises many interesting (and exciting) points and I will try to cover each. The Institute for Space Studies does intend to join the ARPA network, and expects to be operational in the Fall (paperwork and telephone equipment account for the delay). Our purpose is primarily to use the ILLIAC, though we would be delighted to participate actively in other areas as well.

The original presentation of the paper did not include any timings because we felt that they would not be meaningful. In fact, timing was not done until the referee suggested it. The compiler is used locally to permit the mathematicians, physicists, and meteorologists to avoid details of coding and concentrate on problem solving. I have enclosed a copy of the original typescript which contains appendices and other details removed to shorten the paper for publication. The original conclusion (the referee found it too "big headed") is perhaps a better indication of the true value of the compiler. We have also found the compiler to be of use in diagnosis and debugging of programs. An alert is raised when a dead variable is removed, and this is often an indication of clerical error which the production compiler (Fortran-H) does not detect.

I have also enclosed a copy of another paper "A Survey of Compiler Optimization Techniques", which points out (footnote 2) that users either did not exercise or did not understand "explicit control" when once available in Fortran. The use of the "external" statement as in the IBM Fortran-H (extended) compiler will be included and expanded. That is, Fortran-H assumes that certain known functions (e.g., SQRT, SIN) do not

have any side effects, unless they have been listed in an external statement. The extension is to treat all procedures in this manner. The level of optimization (as well as programmer awareness) can be raised significantly if a program must declare possible side effects.

Another aspect in which we hope to continue is in the area of inter-program optimization. If the compiler processes a called program before its caller, it can decide whether or not it is advantageous to include the procedure within the caller. Some work has already been done in this area, in particular when argument values are known the called program may be simplified or perhaps eliminated.

Finally, neither the compiler nor I have any plans of going away. I look forward to your correspondence in the future.

Sincerely yours,

Paul B. Schneck

PBS/is
Encls.: Draft: Fortran to Fortran Optimizing Compiler
A Survey of Compiler Optimization Techniques.